UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/822,115	04/08/2004	Michael C. Stolowitz	NVIDP486/P003259	2463

28875 7590 05/14/2008

Zilka-Kotab, PC
P.O. BOX 721120
SAN JOSE, CA 95172-1120

EXAMINER	
WALTER, CRAIG E	

ART UNIT	PAPER NUMBER
2188	

MAIL DATE	DELIVERY MODE
05/14/2008	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

Commissioner for Patents
United States Patent and Trademark Office
P.O. Box 1450
Alexandria, VA 22313-1450
www.uspto.gov

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

Application Number: 10/822,115
Filing Date: April 08, 2004
Appellant(s): STOLOWITZ, MICHAEL C.

Kevin Zilka
For Appellant

EXAMINER'S ANSWER

This is in response to the appeal brief filed 25 March 2008 appealing from the Office action mailed 26 March 2007.

(1) Real Party in Interest

A statement identifying by name the real party in interest is contained in the brief.

(2) Related Appeals and Interferences

The examiner is not aware of any related appeals, interferences, or judicial proceedings which will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

(3) Status of Claims

The statement of the status of claims contained in the brief is correct.

(4) Status of Amendments After Final

The appellant's statement of the status of amendments after final rejection contained in the brief is correct.

(5) Summary of Claimed Subject Matter

The summary of claimed subject matter contained in the brief is correct.

(6) Grounds of Rejection to be Reviewed on Appeal

The appellant's statement of the grounds of rejection to be reviewed on appeal is correct.

(7) Claims Appendix

The copy of the appealed claims contained in the Appendix to the brief is correct.

(8) Evidence Relied Upon

US Patent	5,765,186	Searby	9 June 1998
US Patent	6,018,778	Stolowitz	25 January 2000
US Patent	5,801,859	Yamamoto	1 September 1998
US PG Publication	2003/0200478 A1	Anderson	23 October 2003

(9) Grounds of Rejection

The following ground(s) of rejection are applicable to the appealed claims:

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

> A person shall be entitled to a patent unless –
>
> (b) the invention was patented or described in a printed publication in this or a foreign country or in public use or on sale in this country, more than one year prior to the date of application for patent in the United States.

Claims 1, 3-5, 8, 10-12, 14, 16, 26-28, and 31 are rejected under 35 U.S.C. 102(b) as being anticipated by Searby (US Patent 5,765,186).

As for claims 1 and 10, Searby teaches a method of reading stored data from an array of independent disk drives so as to provide synchronous data transfer into a buffer, the method comprising:

for each disk drive in the redundant array (Fig. 2 elements 21-24), providing a corresponding FIFO/two-port memory (col. 8, lines 2-6 – RAM buffers (Fig. 2, elements 37-40 are dual port in nature)) for receiving and storing read data responsive to timing signals provided by the respective drive (col. 3, lines 52-67). Also note since data is written in and out sequentially (i.e. the buffers are used to "buffer" data streamed sequentially – i.e. video data - col. 1, lines 7-29), hence the buffer RAMs act as FIFOs;

initiating a READ command to each of the drives of the array, thereby causing each of the drives to retrieve selected elements of its stored data, and to transfer the retrieved data from the drive into its corresponding FIFO/two-port memory using the timing signals provided by the respective drive (col. 5, lines 38-49 – the control bus is used to in conjunction with request signals (REQ) to indicated a READ command – i.e. when data is to be read from the drives to the RAM buffers);

monitoring each of the FIFO/two-port memories to detect a non-empty condition (col. 7, lines 32-46 – the system transfers the data to the data highway (Fig. 2, element 49) based on the determination that data is stored in the RAM buffers. Since all the buffers receive data substantially concurrently (i.e. all disk stores send data to the RAM buffers at the same time, which causes either all the buffers to contain data, or no buffers to contain data), each RAM is monitored to determine that data is either present (i.e. non-empty condition), or not present. Data present in the RAM buffers is an implied acknowledgment that data have

been received from the disk stores. Note the system cannot transfer data to the highway until data is present in the RAM buffers; therefore the memories are monitored for this determination to occur); and

waiting until all of the FIFO/two-port memories indicate such a non-empty condition; then synchronously reading the transferred data from all of the FIFO/two-port memories, thereby forming synchronous read data, and writing the synchronous read data into the buffer (col. 7,lines 32-35 – once all RAMs indicate a non-empty condition (i.e. data is present in each RAM), the data is sent to the data highway and subsequently to the register (i.e. buffer) as shown in Fig. 2, element 50 (see also col. 5, lines 22-37 and col. 6, lines 14-31)). Again, the RAM buffers store data such that either all contain data, or no buffers contain data, therefore the data cannot be transferred until all RAM buffers have data (i.e. are non-empty); and

repeating said monitoring, waiting; reading and writing into the buffer steps until completion of a read operation initiated by the said READ command (col. 7, line 62 through col. 8, line 11 – the system will continue the process provided while the request inputted into the controller is present, the concurrent data transfer will not conclude until the request no longer exists (i.e. completion of the READ command)).

As for claim 26, Searby teaches a method of writing data into an array of independent disk drives, the method comprising:

providing a buffer for storing write data (Fig. 2, element 50);

for each disk drive in the array, providing a corresponding two-port memory for receiving and storing write data (Fig. 2, elements 37-40 – the RAM buffers are dual-port as described in the rejection of claim 1);

monitoring each of the two-port memories to detect a non-full condition (col. 7, lines 32-46 – the system transfers the data to the data highway (Fig. 2, element 49) based on the determination that data is stored in the RAM buffers. Since all the buffers receive data substantially concurrently (i.e. all disk stores send data to the RAM buffers at the same time, which causes either all the buffers to contain data, or no buffers to contain data), each RAM is monitored to determine that data is either present (i.e. non-empty condition), or not present. Note the system cannot transfer data to the highway until data is present in the RAM buffers; therefore the memories are monitored for this determination to occur);

waiting until all of the two-port memories indicate such a non-full condition; then reading write data from the buffer (again the data is read out of the buffer only when data is present in the buffers. The system must detect that data has been written to the RAM buffers before it can transfer data from each of the RAM buffers to the buffer);

computing redundant data from said write data (col. 9, lines 24-59 – redundant data is calculated base on the information in the buffer);

synchronously storing the write data and the computed redundant data into the two-port memories via a first port of each two-port memory (the parity data and the data to be written are stored concurrently, and are stored via the first port of the dual-port memory – col. 9, lines 24-59); and

while synchronously storing the write data and the computed redundant data into the two-port memories, transferring stored data from a second port of each of the two-port memories into its corresponding disk drive, in each case transferring the previously-stored data responsive to timing control provided by the corresponding disk drive (the controller is responsible for asserting the control signal to enable access for writing to and from each buffer RAM simultaneously. Col. 5, lines 50-62, the data can be either written to or from the RAM buffers and disk stores. The memory is dual-port which permits the system to uniquely assign the functions each of the two ports).

As for claim 3, Searby teaches the method of reading stored data from an array according to claim 1 wherein each two-port memory comprises a FIFO memory (data is written in and out sequentially (i.e. the buffers are used to "buffer" data streamed sequentially – i.e. video data - col. 1, lines 7-29), hence the buffer RAMs act as FIFOs).

As for claim 4, Searby teaches the method of reading stored data from an array according to claim 3 wherein the array comprises a redundant array (col. 3, lines 29-37).

As for claim 5, Searby the method of reading stored data from an array according to claim 4 and further comprising, in the event that one of the disk drives of the

redundant array has failed, dynamically regenerating missing data corresponding to the failed drive from the synchronous read data (col. 9, lines 1-39 – parity data is generated and used to replace data in case of a failure).

As for claim 8, Searby teaches the method of reading data from an array according to claim 1 wherein said synchronously reading the stored data from all of the two-port memories comprises asserting a common read enable signal to each of the memories (referring to Fig. 2, the controller (51) asserts the common command line (54) for all RAM buffers to read data from the disk stores).

As for claim 11, Searby teaches the method of reading stored data according to claim 10 wherein the stored data is word striped over the redundant array (col. 4, lines 11-20 and col. 6, lines 14-34 discuss how frames are stored sequentially and striped across each drive. Additionally note the frames are grouped as data words, hence the frames are word striped across the array of disks).

As for claim 12, Searby teaches a method of reading stored data according to claim 10 and further comprising, in the event that one of the disk drives fails to transfer read data to its associated FIFO memory, regenerating the missing read data "on the fly" from the synchronous read data (col. 9, lines 24-60 – the when a error is detected, generated parity information is used account for the error. More specifically, the data regenerator (i.e. data circuitry) performs the bit-wise XOR operation to regenerate missing data due to the drive failure).

As for claim 14, Searby teaches the method of reading data according to claim 10 wherein the synchronous transfer of read data into the common buffer is implemented with a single address counter and a common FIFO read enable signal (referring to Fig. 2, the controller (51) asserts the common command line (54) for all RAM buffers to read data from the disk stores – Additionally, Searby teaches the controller itself as being responsible for generating and counting the addressing to each of the RAM buffers – see Fig. 2).

As for claim 16, Searby teaches a method of reading data from an array according to claim 10 and further comprising providing a FIFO memory in the data path between the individual drive FIFO memories and the common buffer (Fig. 2, elements 4144 depict FIFO buffers between the buffer RAM and the common buffer (50)).

As for claim 27, Searby teaches a method of writing data into an array according to claim 26 and further comprising stalling said storing step whenever any of the two-port memories become full, but only with regard to the full memory, while allowing said synchronously storing the write data to continue into the non-full two-port memories (data storage will continue without delay provided the system recognizes that enough free memory is available in the buffer RAM devices – col. 8, lines 1-11).

As for claim 28, Searby teaches a method of writing data into an array according to claim 27 wherein each two-port memory comprises a FIFO memory (data is written in and out sequentially (i.e. the buffers are used to “buffer” data streamed sequentially – i.e. video data - col. 1, lines 7-29), the buffer RAMs act as FIFOs).

As for claim 31, Searby teaches a method of storing data into an array according to claim 28 wherein said synchronously storing the write data into the FIFOs comprises asserting a common write strobe coupled to all of the FIFO memories (referring to Fig. 2, the controller (51) asserts the common command line (54) for all RAM buffers to read data from the disk stores).

Claim Rejections - 35 USC § 103

The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

> (a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negatived by the manner in which the invention was made.

Claim 15 is rejected under 35 U.S.C. 103(a) as being unpatentable over Searby (US Patent 5,765,186) as applied to claim 10 above.

As for claim 15, though Searby does not explicitly teach a method of reading data from an array according to claim 10 wherein each synchronous transfer of read data into the common buffer stores 64-bits of read data, such a limitation is merely a matter of design choice and would have been obvious in the system of Searby. More specifically, Searby teaches his register (Fig. 2, element 50) as storing information 8-bit format rather than 64-bit as claimed by Applicant. The fact that Searby differs by the claimed invention only by the width of the data stored fails to define a patentably distinct

invention over Searby, since both the claimed invention and Searby's teachings are both directed synchronous data transfer from asynchronous devices.

Claims 6, 7, 13, 29 and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Searby (US Patent 5,765,186) as applied to claims 1, 10, and 26 above and in further view of Anderson (US PG Publication 2003/0200478 A1)

As for claims 6, 7,13, 29, and 30, though Searby teaches all the limitations of claims 1,10, and 26, he failures to teach the read operation as being effected via corresponding UDMA interface to each of the disk drives (or coupled to each of the drives via the UDMA interface as recited in claim 13).

Anderson however teaches a media server with single chip storage controller, which includes a plurality of storage devices (Fig. 2, element 10, interfaced with a controller (12) over a plurality of communication lines (14). Note Anderson specifically teaches the communication lines as being UDMA interfaces (paragraph 0101, all lines)).

It would have been obvious for Searby to further include Anderson's single chip storage controller into his own system for synchronous data transfer. By doing so, Searby would benefit by having a more efficient RAID storage controller which is integrated to enable rapid recovery from failures as taught by Anderson in paragraph 0006, all lines. Anderson specifically advises the application of this benefit to systems such as Searby's, which processes video data. Searby would additionally benefit from

Anderson's reduction of unwanted interruption by enabling hot swapping of failed drives as taught by Anderson in paragraphs 0016 and 0017, all lines.

Claims 2, 9,17 and 19-25 are rejected under 35 U.S.C. 103(a) as being unpatentable over Searby (US Patent 5,765,186) as applied to claims 1, 17, and 26 above and in further view of Stolowitz (US Patent 6,018778).

As for claim 17, Searby teaches an improved RAID disk array controller comprising:

a plurality of disk drive interfaces for attaching physical disk drives (Fig. 2 elements 33-36);

a two-port memory associated with each of the disk drive interfaces (Fig. 2, elements 37-40 – as discussed with claims 1 and 10, the RAM buffers are FIFO in nature and can be dual-port), each two-port memory arranged to store read data provided by the associated disk drive in a disk read operation and, conversely, to provide write data that was previously-stored in the memory to the associated disk drive in a disk write operation (col. 3, lines 52-67 – data is transferred from the disk stores to the RAM buffers. Conversely, the RAMs may write data to the disk store (col. 5, lines 50-62 – this operation may be reversed to allow the RAMs to write to the disk stores));

a logic circuit coupled to all of the two-port memories for detecting when all of the two-port memories have data stored therein for a read operation or available space therein for a write operation (col. 7, lines 32-46 – the system transfers the data to the data highway (Fig. 2, element 49) based on the

determination that data is stored in the RAM buffers. Since all the buffers receive data substantially concurrently (i.e. all disk stores send data to the RAM buffers at the same time, which causes either all the buffers to contain data, or no buffers to contain data), each RAM is monitored to determine that data is either present (i.e. non-empty condition), or not present. Data present in the RAM buffers is an implied acknowledgment that data have been received from the disk stores. Note the system cannot transfer data to the highway until data is present in the RAM buffers; therefore the memories are monitored for this determination to occur (his operation is controlled by the controller (i.e. logic circuit) which is coupled to the RAMs));

control circuitry responsive to the logic circuit for synchronously reading data from all of the two-port memories only when all of the two-port memories have data stored therein, thereby forming synchronous read data (col. 7,lines 32-35 – once all RAMs indicate a non-empty condition (i.e. data is present in each RAM), the data is sent to the data highway and subsequently to the register (i.e. buffer) as shown in Fig. 2, element 50 (see also col. 5, lines 22-37 and col. 6, lines 14-31)). Again, the RAM buffers store data such that either all contain data, or no buffers contain data, therefore the data cannot be transferred until all RAM buffers have data (i.e. are non-empty). The controller, via the aid of SCSI interface logic; controls this function;

the control circuitry further responsive to the logic circuit for detecting that all of the two-port memories have space therein and synchronously writing data

to all of the two port memories thereby forming synchronous write data (the system will continue to buffer the data if it is determined that space is available to stream more data (col. 7, line 62 through col. 8, line 11));

first redundant data circuitry for regenerating missing data "on the fly" from the synchronous read data in the event that one of the disk drives fails to provide read data to its associated two-port memory in a read operation (col. 9, lines 24-60 – the when a error is detected, generated parity information is used account for the error. More specifically, the data regenerator (i.e. data circuitry) performs the bit-wise XOR operation to regenerate missing data due to the drive failure); and

second redundant data circuitry for generating redundant data "on the fly" from the synchronous write data for storing in the array (col. 9, lines 1-39 – the last drive of the array is used to store parity information. In case of a drive failure, the data in this last disk store can be used to generate the data needed to correct the drive failure) and wherein:

Searby however fails to teach one of the disk interfaces as being specifically connected to a redundant disk wherein the first redundant circuitry regenerates corrected in the event of a failed drive without delaying the output of the read data as recited by Applicant in the instant claim.

Stolowitz however teaches a disk array controller including:

one of the disk drive interfaces is designated for connection to a redundant disk drive for storing redundant data so that the synchronous read

data includes redundant data (col. 13, line 55 through col. 14, line 3 – the drives include a redundant drive) ;

the first redundant data circuitry for regenerating missing data is coupled to all of the memories to receive the synchronous read data, including the redundant data, and further is arranged to compute a redundant data operation across the synchronous read data so as to provide corrected data in the event of a failed drive without delaying output of the read data (col. 8, lines 25-65 – once a drive fails, the system can reconstruct the lost data, and continue to perform the necessary step for data transfer by excluding the drive that failed. Note Stolowitz explicitly teaches no delay occurring during this process).

It would have been obvious to one of ordinary skill in the art at the time of the invention for Searby to further include Stolowitz's disk array controller and method to synchronously transfer data into his own system of synchronously transferring data. By doing so, Searby would benefit by improving his own disk array performance and improve his disk array storage reliability while reducing the cost and complexity of his present controller to perform a synchronous data transfer as taught by Stolowitz (col. 6, lines 4-18).

As for claim 19, Searby teaches an improved RAID disk array controller according to claim 17 wherein each two-port memory comprises a FIFO memory data is written in and out sequentially (i.e. the buffers are used to "buffer" data streamed sequentially – i.e. video data - col. 1, lines 7-29, the buffer RAMs act as FIFOs).

As for claim 21, Searby teaches an improved RAID disk array controller according to claim 17 and further comprising a single address counter arranged for addressing the buffer for transfers between the buffer and the FIFO memories in either direction (the controller itself is responsible for generating and counting the addressing to each of the RAM buffers – see Fig. 2 – Additionally note the data transfer can occur either from disk stores to the RAM buffers or vise versa – col. 5, lines 50-62).

As for claim 9, though Searby teaches the method of reading data from an array according to claim 1, he fails to teach wherein said synchronously reading the stored data from all of the two-port memories is conducted over a single DMA channel.

Stolowitz however teaches a disk array controller that synchronously transfers data between data port and a buffer memory, which uses a single DMA channel to interface between the drives and the memories (col. 11, lines 5-24).

As for claim 2, Stolowitz teaches the stored data includes as including user data as well as redundant data sufficient to enable reconstruction of all of the user data in the event of a failure of any single drive of the array and the method further comprising, in the event that one of the disk drives fails, executing said initiating, monitoring, waiting and synchronously reading steps only with respect to the non-failed drives; and dynamically regenerating missing data corresponding to the failed drive from the synchronous read data (col. 8, lines 25-65 – once a drive fails, the system can reconstruct the lost data, and continue to perform the necessary step for data transfer by excluding the drive that failed).

As for claim 20, Stolowitz teaches a buffer comprising DRAM (Fig. 6, element 106).

As for claims 22-23, Stolowitz teaches all drives implement a ATA/ATAPI interface (Fig. 2, all drives are IDE – Stolowitz discusses the equivalence of IDE and ATA – col. 4, lines 1-10). Note this similar comparison is discussed in Applicant's original specification paragraph 0005.

As for claim 24, Stolowitz teaches the array controller as being implemented on a signal motherboard (col. 13, line 57 through col. 14, line 3).

As for claim 25, Stolowitz teaches the array controller as being implemented on a Host Bus Adapter (Fig. 2, element 104 - since a host bus adapter is defined as "a device for connecting a peripheral to the main computer", the host interface serves as a host bus adapter as it interfaces the array controller and disks with the host itself via the host bus (Fig. 2, element 102)).

It would have been obvious to one of ordinary skill in the art at the time of the invention for Searby to further include Stolowitz's disk array controller and method to synchronously transfer data into his own system of synchronously transferring data. By doing so, Searby would benefit by improving his own disk array performance and improve his disk array storage reliability while reducing the cost and complexity of his present controller to perform a synchronous data transfer as taught by Stolowitz (col. 6, lines 4-18).

Claim 18 is rejected under 35 U.S.C. 103(a) as being unpatentable over the combined teachings of Searby (US Patent 5,765,186) and Stolowitz (US Patent 6,018,778) as applied to claim 17 above, and in further view of Yamamoto (US Patent 5,801,859).

As for claim 18, though the combined teachings of Searby and Stolowitz disclose, allowing the dual port memory to transfer data in either direction via unique I/O ports, they fail to teach exchanging them via multiplexers as claimed by Applicant. Yamamoto however teaches a network system for plural node devices without arbitration, in which the port (either input or output) can be exchanged via multiplexing logic (col. 30, lines 3-18).

It would have been obvious to one of ordinary skill in the art at the time of the invention for the combined teachings of Searby and Stolowitz to further include Yamamoto's system into his own system for synchronously transferring data. By doing so, they would benefit by having a more efficient means of data transfer while preventing the need for increased hardware costs as taught by Yamamoto in col. 5, lines 51-67.

(10) Response to Argument

Issue #1: 1 Group

Group #1: 1 Argument

Argument #1 (page 12, ll. 4-13 of the Appeal Brief filed 25 March 2008, hereinafter 'Brief'): Examiner fully considered Appellant's argument, and it is found to be persuasive. Examiner wishes to withdraw this rejection, hence removing the issue for consideration under appeal.

Issue #2: 1 Group

Group #1: 1 Argument

Argument #1 (page 12, l. 14 through page 13, l. 6 of the Brief): Examiner fully considered Appellant's argument, and it is found to be persuasive. Examiner wishes to withdraw this rejection, hence removing the issue for consideration under appeal.

Issue #3: 2 Groups

Group #1: 3 Arguments

Argument #1 (page 13, l. 7 through page 15, l. 24 of the Brief): Appellant asserts, Searby “does not teach ‘receiving and storing read data responsive to timing signals provided by the respective drive’” Additionally, Appellant asserts that Searby fails to teach receiving and storing read data responsive to timing signals provided by the respective drive. Appellant evidences these allegations by contrasting Examiner’s application of Searby with the instant claims, stating “[t]hus, Searby's mere disclosure of waiting until all of the disc interfaces have data ready before transferring data fails to meet appellant's specific claim language, namely "memory for receiving read data responsive to timing signals provided by the respective drive" in addition to "synchronously reading the transferred data from all of the...memories". Similar arguments continue on page 14, l. 17 through page 15, l. 24 of the Brief whereby Appellant contrasts cited portions of Searby relied upon by Examiner with the instant claims. It appears based on these arguments that the point on contention between Appellant and Examiner is whether or not the request signals utilized by the drives via their respective interfaces in fact constitutes “timing signals provided by the respective drive[s]”. Appellant essentially concludes, “merely waiting until all of the disc interfaces have data ready before transferring data, as in Searby, does not teach "transfer[ring] the retrieved data from the drive into its corresponding two-port memory using the timing provided by the respective drive".

These arguments however are not persuasive. Referring to the previously cited col. 5, ll. 38-49) Examiner clearly mapped each of the drive's associated respective signals, to Appellant's timing information (see also Examiner response to arguments in the Office action made FINAL 26 March 2007 – page 20, ll. 1-7). To provide further clarification, Examiner notes that in Searby, it is the controller that in fact waits on each drive's respective SCSI interfaces to output their respective signals until the data is transferred to each of their respective RAMs (see col. 6, l. 59 through col. 7, l. 9). Examiner maintains that this "waiting" for an outputted request as disclosed by Searby is in fact necessary for data to be properly transferred from each drive to its respective RAM, and further constitutes "timing information" because the request is used for the express purpose of ensure data synchronization via correct timing. Each drive's SCSI interface is merely a conduit for each of the drive's data and control information, therefore these requests (e.g., timing information) must in fact be "provided by" each drive.

Argument #2 (page 15, l. 25 through page 17, l. 23 of the Brief): Appellant asserts, “merely writing data into all of the RAM buffers using a common strobe signal, in addition to providing RAM addressing, as taught in Searby, fails to disclose ‘monitoring each of the two-port memories to detect a non-empty condition’”. Further Appellant contends, “as all of the RAMs receive data concurrently, monitoring any one of them would suffice to determine its status. Since the process in Searby is all synchronous, the controller has no need to "monito[r] each of the two-port memories to detect a non-empty condition". Appellant concludes by citing a portion of Searby not explicitly relied upon by Examiner (namely, col. 7, ll. 39-42) and attempts to rebut Examiner’s inherency argument via a description of Searby’s tri-state buffers (again, not relied upon by Examiner in the rejection). Further, Appellant fails to address critical lines of Searby relied upon by Examiner to form the rejection (namely, col. 7, ll. 32-38 – these lines clearly disclose that data in the RAM can be outputted only when data is in all RAM buffers (e.g. non-empty condition)). It appears based on these arguments that the two points of contention between Appellant and Examiner are (1) whether or not “monitoring each of the two-port memories to detect a non-empty condition” is inherent to Searby’s system, and (2) whether or not Searby’s system “waits” until all two-port memories are “non-empty” prior to outputting the data.

These arguments however are not persuasive. Examiner maintains pursuant to MPEP § 2112 (IV.), “[t]o establish inherency, the extrinsic evidence ‘must make clear that the missing descriptive matter is necessarily present in the thing described in the reference, and that it would be so recognized by persons of ordinary skill. Inherency, however,

may not be established by probabilities or possibilities. The mere fact that a certain thing may result from a given set of circumstances is not sufficient.' " In re Robertson, 169 F.3d 743, 745, 49 USPQ2d 1949, 1950-51 (Fed. Cir. 1999)". Examiner maintains with respect the aforesaid first point of contention that memory monitoring is inherent in a system such as Searby's. More specifically, before Searby's system is capable of transferring data to the data highway and ultimately to the buffer (col. 7, lines 32-35), the system must in fact determine that data is stored in each of the buffers (since Searby's system is directed to the parallel transfer of data, the system must wait until each of the memories has data present before the data can be sent to the highway). A system of Searby's would clearly not be enabled if it were not cognizant of data stored in each of its RAM buffers, hence this "monitoring" is in fact inherent. Searby must wait until data is present before it can be stored, therefore his system must inherently perform some sort of monitoring process of the memories to determine if data is present to transfer (i.e. non-empty condition). Since Examiner has no intrinsic evidence from Appellant specification to rely upon to further define "monitoring" (i.e. a definition of "monitoring" is not explicitly addressed within the specification), Examiner is forced to rely upon extrinsic evidence for which to interpret the term "monitoring". One of ordinary skill in the art would understand "monitoring" to include "the observation, detection, or recording of an operation of a machine or system" (standard English dictionary definition), therefore Examiner has properly applied this definition against the claims as being the "broadest reasonable interpretation consistent with Application specification" pursuant to MPEP § 2111. Since Searby's system must inherently observe or detect if

data is stored in each memory before said data can be transferred (otherwise the system would be incapable of accessing the data stored therein), Searby in fact anticipates this limitation. With respect to the aforesaid second point of contention, Examiner again points to the cited lines not contemplated by Appellant when addressing Examiner's rejection (col. 7, ll. 32-38) which clearly describe waiting for data to be present in all RAMs prior to transmission. The lines additional address Appellant's concern that only one RAM would be monitored if data is transferred to them (system must wait until data is present in all RAMs).

Argument #3 (page 17, l. 24 through page 19, l. 4 of the Brief): Appellant asserts, "merely waiting to collect a full frame of video data, where the data was already synchronized as it was written into the RAM buffer, fails to disclose "synchronously reading data from all of the two-port memories...thereby forming synchronous read data"". Further, Appellant contends, "the mere disclosure that the word is output as each tri-state buffer is enabled in sequence and individually output from the register to the highway, as in Searby, simply fails to even suggest "forming synchronous read data" by "synchronously reading the transferred data from all of the two-port memories"".

These arguments however are not persuasive. First, the entire purpose of Searby is to synchronize and buffer image data for parallel and concurrent data transfer (abstract and col. 4, ll. 11-28). Referring again to col. 3, ll. 32-35, by "waiting" until all RAMs contain data, and transmitting data from all RAM concurrently, Searby is in fact "forming synchronous read data" and "reading [said] data from all of the two-port memories" as required by the claim. As such, Searby renders these limitations anticipated, Appellant's arguments notwithstanding.

Group #2: 1 Argument

Argument #1 (page 19, l. 5 through page 19, l. 16 of the Brief):

Appellant's arguments are identical to those set forth under Issue #3, Group #1, Argument #2, therefore those arguments are not persuasive for the reasons stated by Examiner under that heading, *supra*.

Issue #4: 1 Group

Group #1: 1 Argument

Argument #1 (page 19, l. 17 through page 21, l. 11 of the Brief):

Appellant's asserts that even though Searby teaches at least 16-bit words reduced to 8-bit portions, storing 64-bits of read data as claimed by Appellant is in fact non-obvious in a system of Searby's, counter to Examiner's assertion otherwise. Appellant further formally requests a specific showing of the subject matter in ALL of the claims in any future action, and contends that Searby in fact fails to teach or suggest all the claim limitations.

This argument however is not persuasive. Examiner wishes to underscore the fact that this obviousness rejection asserted by Examiner was NOT in fact traversed by Appellant in response to the non-final action mailed 13 July 2006. In fact, Appellant explicitly conceded that claim 15 is not in fact, by itself, independently patentable ("Applicant does not contend that claim 15 is independently patentable; it is patentable for the reasons explained above with regard to claim 10, as amended" – see Applicant's remarks, 14 January 2007).

Since Appellant explicitly conceded that claim 15 is NOT independently patentable, and he or she did not traverse this Official notice while the application was under non-final rejection, this limitation is in fact considered admitted prior art pursuant to MPEP § 2144.03 (paragraph C).

Issue #5: 2 Groups

Group #1: 1 Argument

Argument #1(page 21, ll. 12-21 of the Brief)

Appellant's arguments are identical to those set forth under Issue #3, Group #1, therefore those arguments are not persuasive for the reasons stated by Examiner under that heading, *supra*.

Group #2: 1 Argument

Argument #1 (page 21, ll. 12-21 – "mislabeled Group #1" of the Brief)

Appellant's arguments are identical to those set forth under Issue #3, Group #2, therefore those arguments are not persuasive for the reasons stated by Examiner under that heading, *supra*.

Issue #6: 2 Groups

Group #1: 1 Argument

Argument #1 (page 22, ll. 4-6 of the Brief)

Appellant's arguments are identical to those set forth under Issue #3, Group #1, therefore those arguments are not persuasive for the reasons stated by Examiner under that heading, *supra*.

Group #2: 1 Argument

Argument #1 (page 22 l. 7 through page 23 l. 14 of the Brief)

Appellant's arguments are identical to those set forth under Issue #3, Group #2, Argument #3, therefore those arguments are not persuasive for the reasons stated by Examiner under that heading, *supra*.

Issue #7: 1 Group

Group #1: 1 Argument

Argument #1 (page 23, ll. 15-23 of the Brief)

Appellant's arguments are identical to those set forth under Issue #6, Group #2, therefore those arguments are not persuasive for the reasons stated by Examiner under that heading, *supra*.

(11) Related Proceeding(s) Appendix

No decision rendered by a court or the Board is identified by the examiner in the Related Appeals and Interferences section of this examiner's answer.

For the above reasons, it is believed that the rejections should be sustained.

Respectfully submitted,

/Craig E Walter/
Patent Examiner, Art Unit 2188

Conferees:

/Hyung S SOUGH/
Supervisory Patent Examiner, Art Unit 2188
05/12/08

/Kevin L Ellis/
Acting Supervisory Patent Examiner, Art Unit 2187
5/12/08